

# CSE211 Digital Design

Akdeniz University

Week12: Sequential Logic Part 1

1

Assoc.Prof.Dr. Taner Danişman

tdanisman@akdeniz.edu.tr

# Course program

2

Week 01	09/16/2024	Introduction
Week 02	09/23/2024	Digital Systems and Binary Numbers I
Week 03	09/30/2024	Digital Systems and Binary Numbers II
Week 04	10/07/2024	Boolean Algebra and Logic Gates I
Week 05	10/14/2024	Boolean Algebra and Logic Gates II
Week 06	10/21/2024	Gate Level Minimization
Week 07	10/28/2024	Karnaugh Maps
Week 08	11/04/2024	Karnaugh Maps
Week 09	11/11/2024	Midterm
Week 10	11/18/2024	Combinational Logic
Week 11	11/25/2024	Combinational Logic
Week 12	12/02/2024	Timing, delays and hazards
Week 13	12/09/2024	Synchronous Sequential Logic
Week 14	12/16/2024	Synchronous Sequential Logic

# Synchronous Sequential Logic

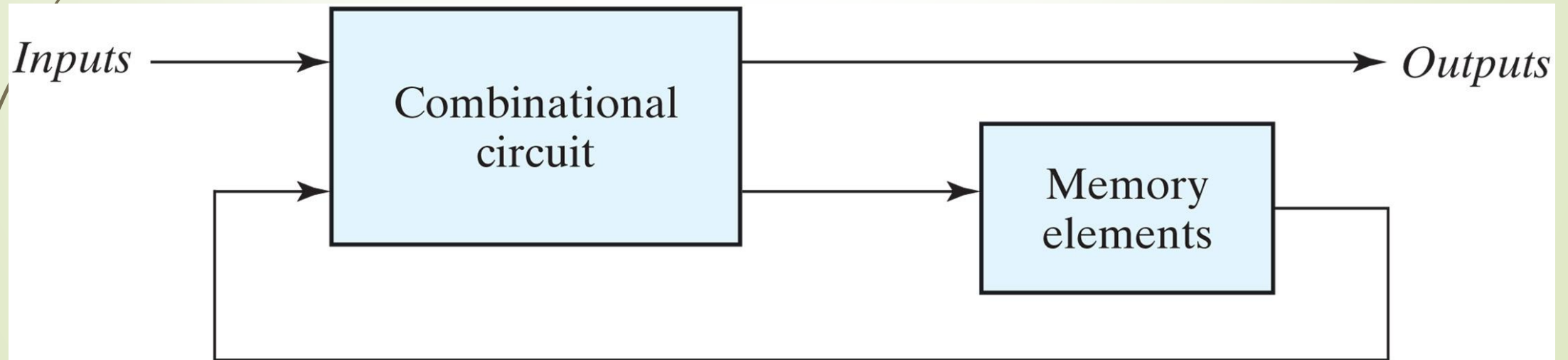
3

- In combinational logic the output depends only and immediately on their inputs.
- A combinational logic cannot have some sort of **feedback** loop where the output of a logic is also the input of it.
- A **sequential network** on the other hand can have this feature.
- All consumer products, cell phones, navigation receivers, personal computers, digital cameras have the ability to send, receive, store, retrieve, and process information represented in a **binary format**.
- This is only possible with a special electronic component which can store information, **the memory**..

# Block Diagram of a Sequential Circuit

4

- The binary information stored in **memory elements** at any given time defines the **state** of the sequential circuit at that time.
- External inputs also determine the condition for changing the state in the storage elements
- **A sequential circuit is specified by a time sequence of inputs, outputs, and internal states**



# Types of Sequential Circuits

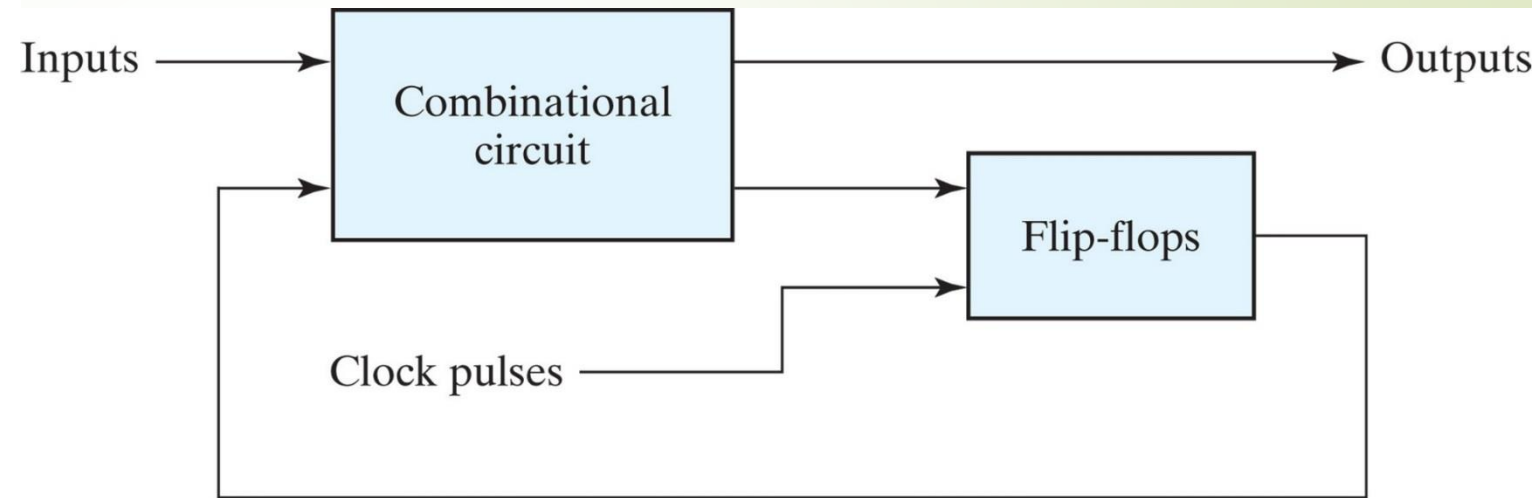
5

- **Synchronous** sequential circuit
  - whose behavior can be defined from the knowledge of its signals at discrete instants of time.
  - Synchronization is achieved by a **timing device** called a **clock generator**
    - denoted by the identifiers **clock** and **clk**
  - Synchronous sequential circuits that use clock pulses to control storage elements are called **clocked sequential circuits**
  - **Less instable than asynchronous systems.**
- **Asynchronous** sequential circuit
  - behavior of an *asynchronous* sequential circuit depends upon **the input signals** at any instant of time *and* the **order** in which the inputs change
  - The storage elements commonly used in asynchronous sequential circuits are time-delay devices
  - the **internal propagation delay of logic gates** is of **sufficient** duration to produce the needed delay, so that actual delay units may not be necessary.
  - Thus, an asynchronous sequential circuit may be regarded as a **combinational circuit with feedback**.

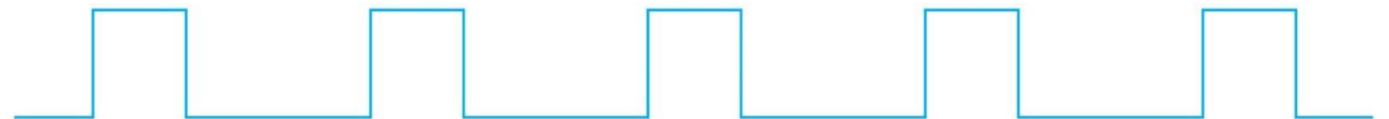
# Synchronous sequential circuits

6

- The storage elements (memory) used in clocked sequential circuits are called **flipflops**.
- A flip-flop is a binary storage device capable of storing **one bit of information**
- **In a stable state**, the output of a flip-flop is either **0** or **1**
- The **new value** is stored (i.e., the flip-flop is updated) when a pulse of the clock signal occurs



(a) Block diagram



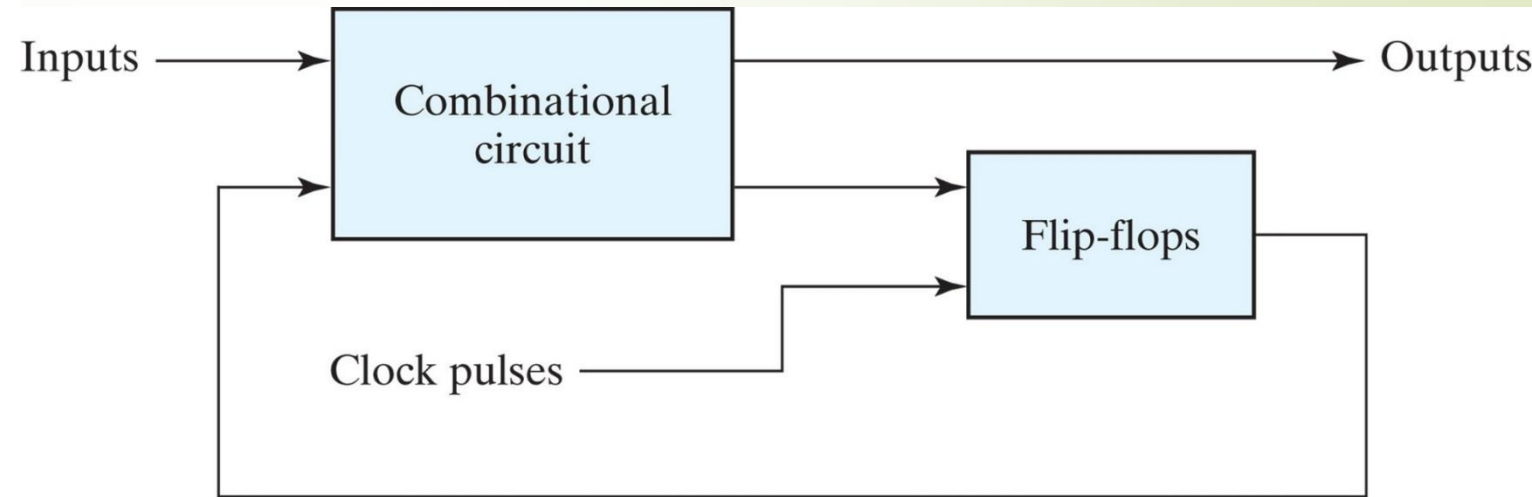
(b) Timing diagram of clock pulses



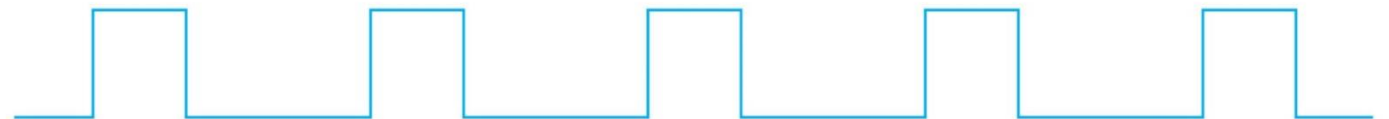
# Synchronous sequential circuits

7

- Prior to the occurrence of the clock pulse, the combinational logic forming the next value of the flip-flop must have reached a **stable value**.
- Consequently, the **speed** at which the **combinational logic circuits** operate is critical
- the combinational logic must respond to a change in the state of the flip-flop in time to be updated before the next pulse arrives.



(a) Block diagram



(b) Timing diagram of clock pulses

## 5.3 Storage Elements: Latches

8

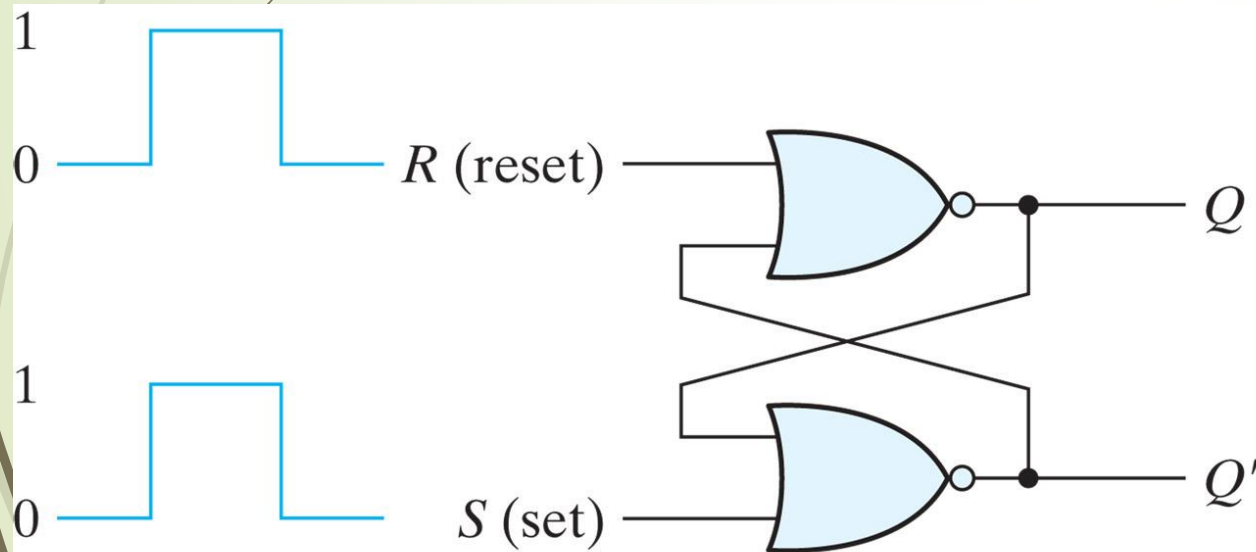
- Latches: Storage elements that **operate with signal levels** (rather than signal transitions) are referred to as **latches** ;
  - Latches are said to be **level sensitive devices**.
  - Useful in asynchronous sequential circuits
  - Not practical for use as storage elements in synchronous sequential circuits
- Flip-flops: Those controlled by a **clock** transition are **flip-flops**.
  - flip-flops are **edge-sensitive devices**.



# SR Latch (Active High)

9

- The SR latch is a circuit with **two cross-coupled NOR gates** or **two cross-coupled NAND gates**, and **two inputs** labeled **S** for **set** and **R** for **reset**.



(a) Logic diagram

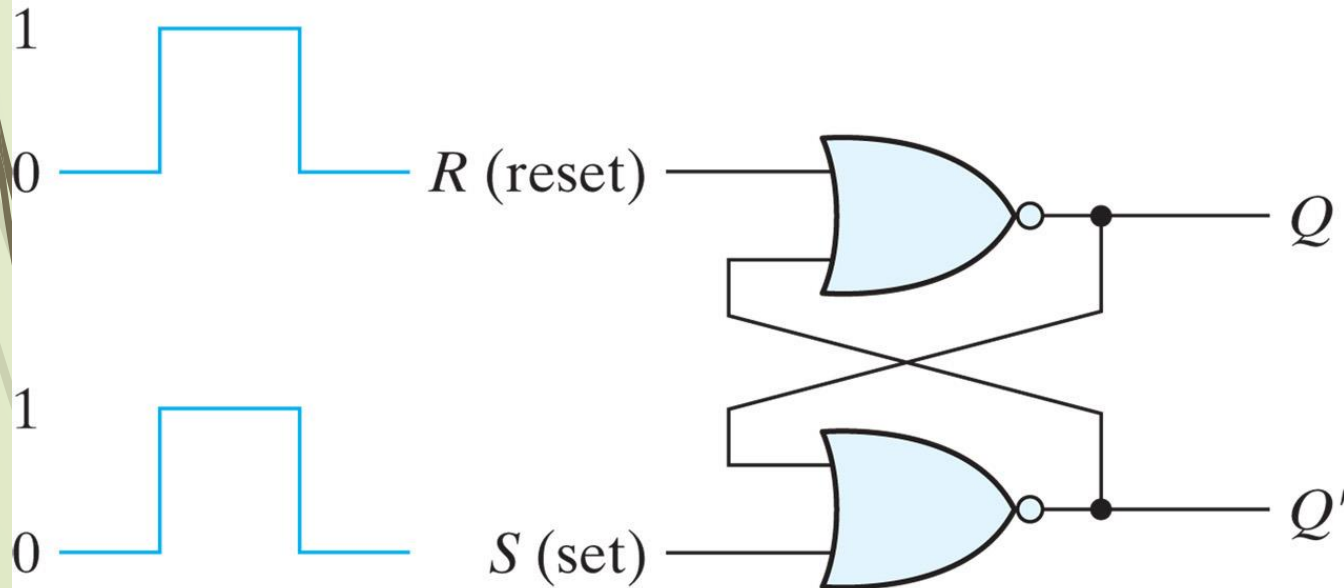
$S$	$R$	$Q$	$Q'$	
1	0	1	0	
0	0	1	0	(after $S = 1, R = 0$ )
0	1	0	1	
0	0	0	1	(after $S = 0, R = 1$ )
1	1	0	0	(forbidden)

(b) Function table

# SR Latch with NOR gate

10

- The latch has two useful states.
- When output  $Q = 1$  and  $Q' = 0$ , the latch is said to be in the **set state**
- When  $Q = 0$  and  $Q' = 1$ , it is in the **reset state**
- Under normal conditions, both inputs of the latch remain at 0 unless the state has to be changed.
- The application of a momentary **1** to the **S** input causes the latch to go to the set state
- **Removing the active input from S leaves the circuit in the same state**

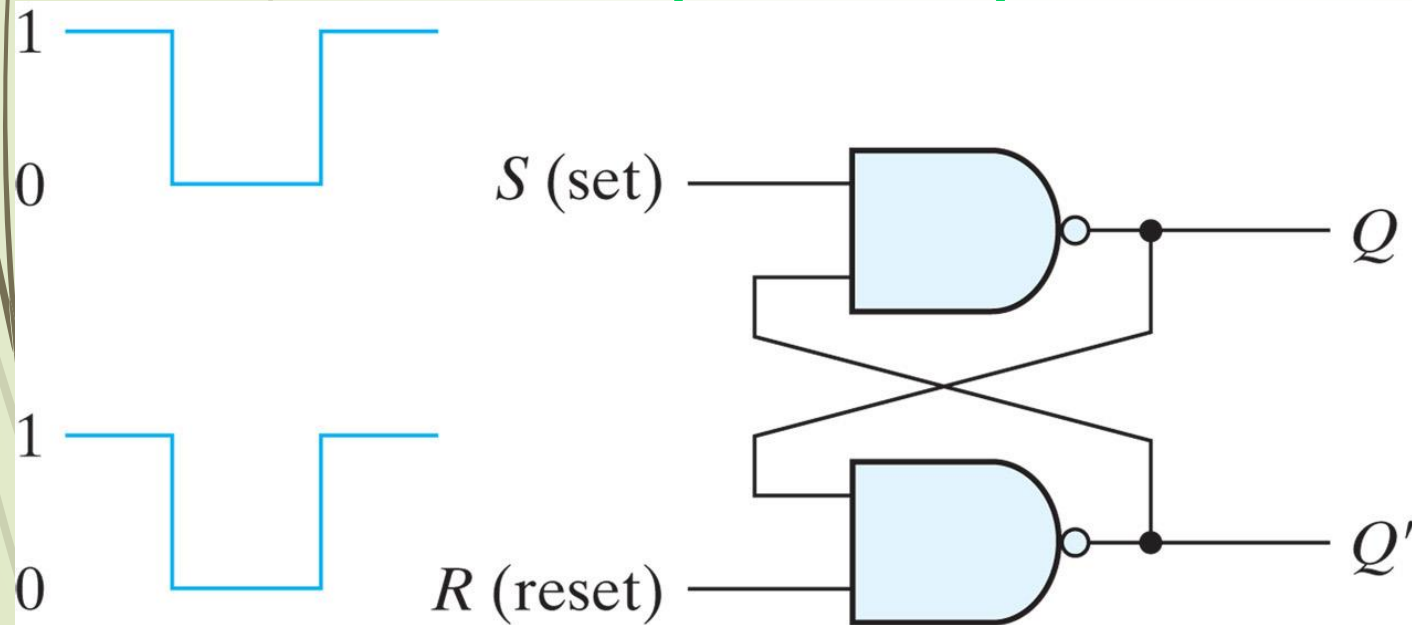


$S$	$R$	$Q$	$Q'$	
1	0	1	0	
0	0	1	0	(after $S = 1, R = 0$ )
0	1	0	1	
0	0	0	1	(after $S = 0, R = 1$ )
1	1	0	0	(forbidden)

# SR Latch with NAND gate (Active Low)

11

- Because the NAND latch requires a 0 signal to change its state, it is sometimes referred to as an  $S'R'$  latch
- **Video-1** <https://www.youtube.com/watch?v=-aQH0ybMd3U>



(a) Logic diagram

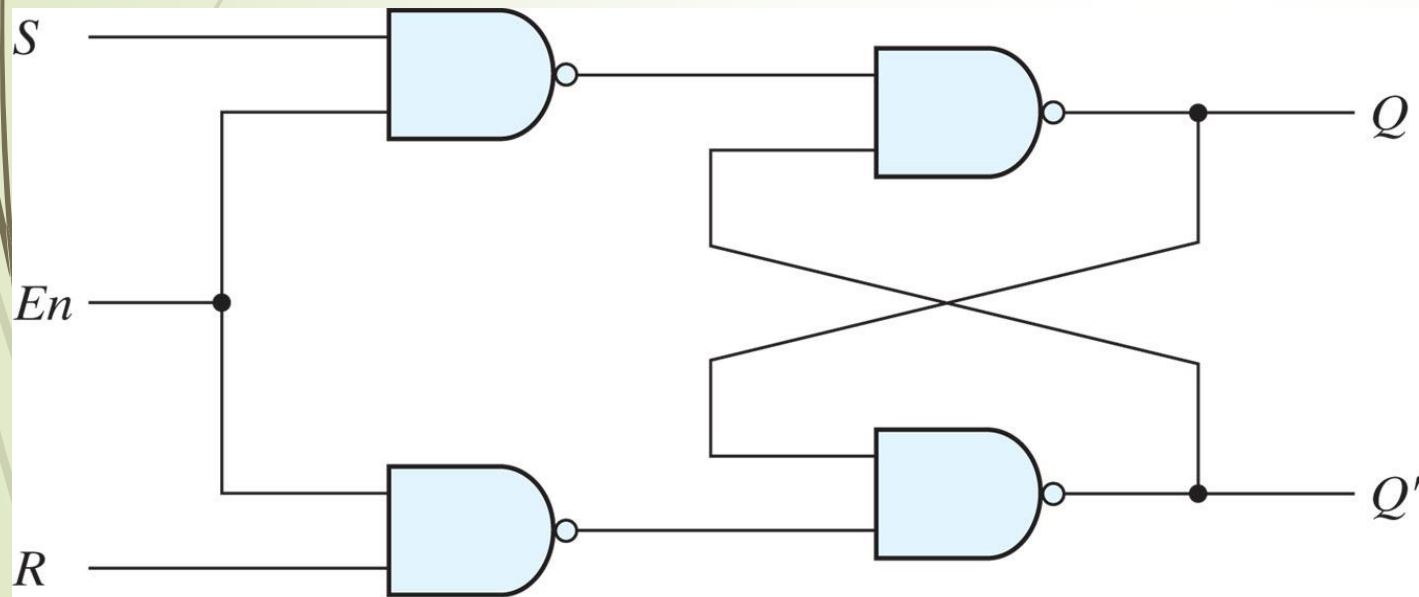
$S$	$R$	$Q$	$Q'$	
1	0	0	1	
1	1	0	1	(after $S = 1, R = 0$ )
0	1	1	0	
1	1	1	0	(after $S = 0, R = 1$ )
0	0	1	1	(forbidden)

(b) Function table

# SR latch with control input (Gated)

12

- The outputs of the NAND gates stay at the logic-1 level as long as the enable signal remains at 0.
- Video-2  
[https://www.youtube.com/watch?v=HxAhOETcwr4&ab\\_channel=ComputerScience](https://www.youtube.com/watch?v=HxAhOETcwr4&ab_channel=ComputerScience)



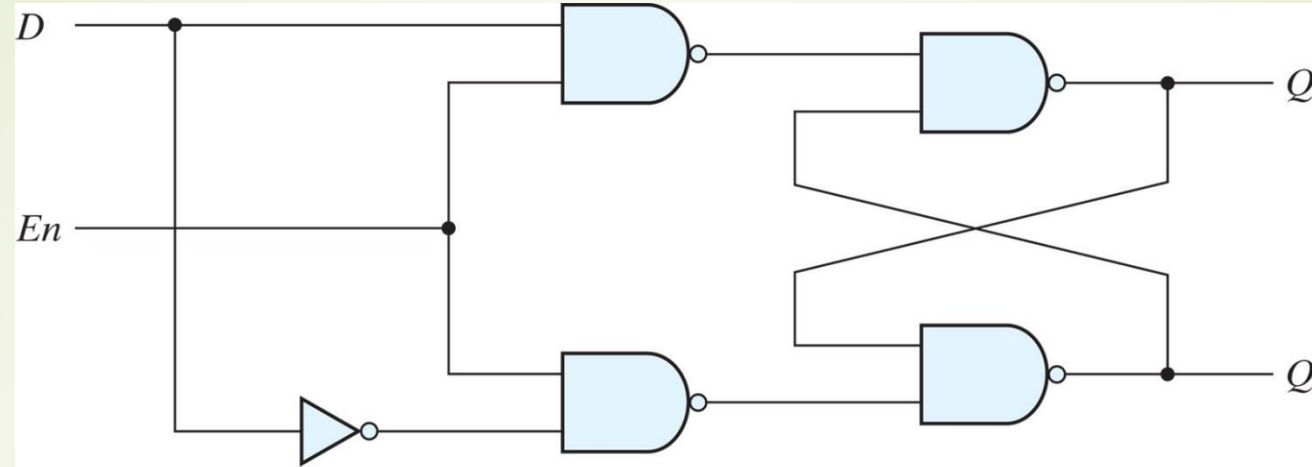
(a) Logic diagram

$En$	$S$	$R$	Next state of $Q$
0	X	X	No change
1	0	0	No change
1	0	1	$Q = 0$ ; reset state
1	1	0	$Q = 1$ ; set state
1	1	1	Indeterminate

(b) Function table

# D Latch (Transparent Latch)

13



(a) Logic diagram

$En$	$D$	Next state of $Q$
0	X	No change
1	0	$Q = 0$ ; reset state
1	1	$Q = 1$ ; set state

(b) Function table

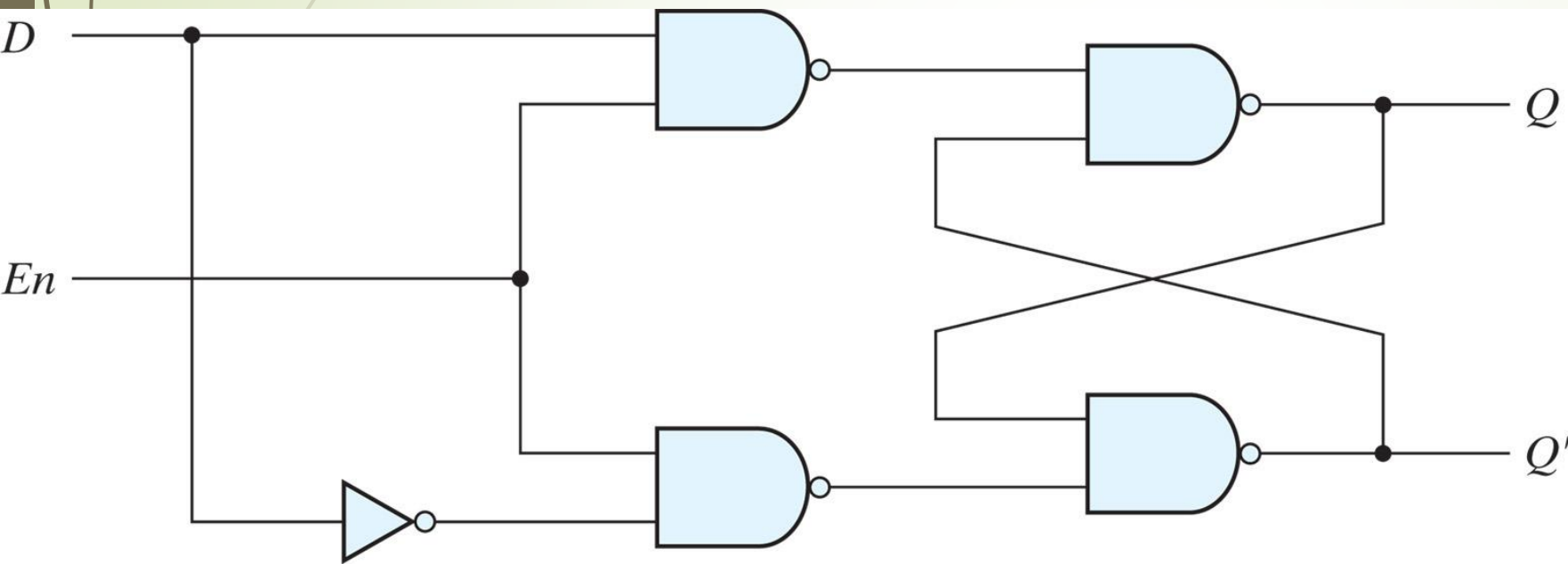
Copyright ©2013 Pearson Education, publishing as Prentice Hall

- It is suited for use as a temporary storage for **binary information**
- The **binary information** present at the data input of the  $D$  latch is **transferred to the  $Q$  output** when the **enable input is asserted**
- The **output follows changes** in the data input as long as the enable input is asserted
- For this reason, the circuit is often called a **transparent latch**
- When the enable input signal is de-asserted, the **binary information** that was present at the data input at the time the transition occurred **is retained**

# D Latch (Transparent Latch)

14

- One way to eliminate the undesirable condition of the indeterminate state in the  $SR$  latch is to ensure that inputs  $S$  and  $R$  are never equal to 1 **at the same time**
- Video-3 [https://www.youtube.com/watch?v=y7Zf7Bv\\_J74](https://www.youtube.com/watch?v=y7Zf7Bv_J74)



(a) Logic diagram

$En$	$D$	Next state of $Q$
0	X	No change
1	0	$Q = 0$ ; reset state
1	1	$Q = 1$ ; set state

(b) Function table

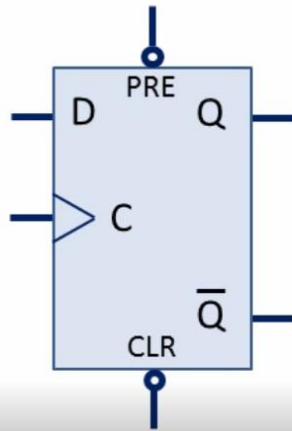


# Clocked D Latch

15

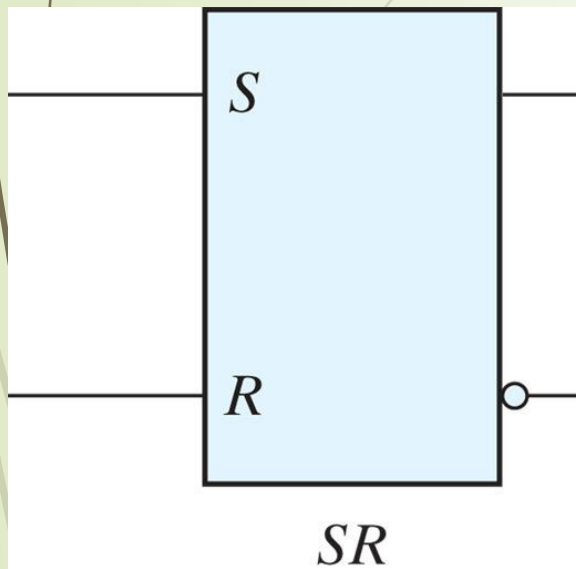
- Controlling over the behavior of the latch
- **Video 4 - <https://www.youtube.com/watch?v=8bUKw2cGcGg>**

## Clocked D Latch



# Different Latch Symbols

16



## 5.4 Storage Elements: Flip-Flops

3

- The state of a latch or flip-flop is switched by a change in the control input. This momentary change is called a *trigger*.
- The problem with the latch is that it responds to a change in the level of a clock pulse. Glitches are problematic.

